

Sistema de control de luz a partir de comandos de voz usando RNA

Sergio Velásquez, Efraín Oviedo, Claudia Isaza
sergiovr12@gmail.com, eaoc46@gmail.com, cisaza@udea.edu.co
Departamento de Ingeniería Electrónica, Universidad de Antioquia
Medellín - Colombia

Resumen—En este artículo, se presenta un sistema que permite controlar, por voz, la intensidad de luz de una lámpara. Éste sistema tiene la característica de no requerir hardware especializado permitiendo un bajo costo de implementación. Para el reconocimiento de voz se realiza una comparación del desempeño usando máquinas de soporte vectorial y redes neuronales, de tipo perceptron y de base radial. Para verificar la generalización de cada alternativa, se utiliza la técnica de *cross validation*. El sistema es capaz de reconocer cuatro comandos de voz pronunciados por personas de cualquier género y edad, y el objetivo es ejercer el control de intensidad de luz de una lámpara.

Palabras clave- Reconocimiento de voz; Redes Neuronales; Máquinas de soporte vectorial; Control de intensidad lumínica; Domótica.

I. INTRODUCCIÓN

La domótica es un área de interés puesto que es útil para proveer comodidades, seguridad, entretenimiento, entre otros aspectos, que contribuyen a mejorar la calidad de vida de las personas. Se define como el conjunto de soluciones que, usando las técnicas y tecnologías disponibles, logran una mejor utilización, gestión y control de todos los aspectos relacionados con la vivienda [1]. Dentro de las aplicaciones en esta área se encuentra el control de alarmas, control de apertura y cierre de persianas, además del control de la intensidad de luz de las lámparas. Es precisamente en este último tema donde se centra éste artículo, se presenta un sistema capaz de controlar la intensidad lumínica de una lámpara mediante la voz humana.

El reconocimiento de comandos de voz le permite al hombre comunicarse, de una manera sencilla, con las máquinas [2]. Este tema ha sido motivo de estudio en múltiples ocasiones. Ramírez presenta, en 2008, un sistema que está en capacidad de reconocer una palabra pronunciada por un mismo usuario [3], en este estudio se utilizan las Redes Neuronales ([4, 5, 6]) como técnica de reconocimiento. Usando esta misma técnica, Moreno et al, propone un sistema que permite reconocer tres palabras distintas, pronunciadas por un mismo hablante [7]. También se han diseñado sistemas de reconocimiento de voz para identificar las vocales [8] y los números del 1 al 5 [9].

En la actualidad existen sistemas domóticos que permiten controlar la intensidad lumínica de una lámpara mediante

aplausos, control remoto, pantalla táctil, internet, línea telefónica, entre otros.

En los estudios mencionados anteriormente, se han obtenido resultados óptimos, pero el objetivo no ha sido el reconocimiento de una palabra pronunciada por cualquier persona, lo cual es necesario para aplicaciones de domótica. La propuesta presentada en este artículo permite identificar cuatro comandos de voz pronunciados por personas de cualquier sexo y edad; esto implica que no se reconoce una voz o un grupo de voces en particular. Por tratarse de una aplicación que va a ser usada por todos los miembros de una casa, el sistema está en capacidad de responder asertivamente a la voz de cualquier persona incluso si no ha participado anteriormente en el proceso de recolección de muestras.

El documento se divide en las siguientes secciones: en la sección II se explica el tratamiento que se hace a la señal de audio, en tiempo y en frecuencia, antes de realizar el entrenamiento del sistema clasificador; la sección III es dedicada al sistema clasificador, se compara el desempeño de dos tipos de redes neuronales, redes de base radial RBF y perceptron; también se comentan los resultados usando máquinas de soporte vectorial SVM; la sección IV presenta los componentes de software y hardware necesarios para el funcionamiento del sistema. En la sección V se presentan los resultados obtenidos del prototipo implementado y finalmente, la sección VI es dedicada a las conclusiones y perspectivas.

II. TRATAMIENTO SEÑAL DE AUDIO

Para controlar la intensidad de luz de la lámpara se escogieron cuatro comandos básicos que el usuario podrá dictar y el sistema debe estar en capacidad de reconocer. Dichos comandos son: Encender, Apagar, Subir, Bajar.

El comando encender, permite prender la lámpara al 100%, el comando apagar permite apagar la lámpara por completo, y los comandos subir y bajar realizan un cambio equivalente al 25% en el nivel de intensidad actual.

Cuando una palabra es pronunciada, se guarda usando una tarjeta de sonido. La frecuencia de grabación utilizada influye en la calidad de las muestras y en la cantidad de datos que se tienen para determinar el comando pronunciado. La grabación se realizó con una frecuencia de muestreo relativamente alta (44.1KHz) y posteriormente se le aplicaron varios filtros para

analizar la cantidad de componentes de la señal que se pierden al bajar la frecuencia de muestreo.

En el pretratamiento de las muestras de audio en el dominio del tiempo, se aplica un filtro pasa bajas a una frecuencia de 4KHz de los archivos remuestreados a 8KHz, para conservar solo los elementos representativos de la voz. También se hace un recorte de los archivos de audio eliminando los silencios más prolongados.

Una comparación de una muestra de audio, antes y después de aplicarle los cambios anteriormente mencionados, se presenta en la Figura 1.

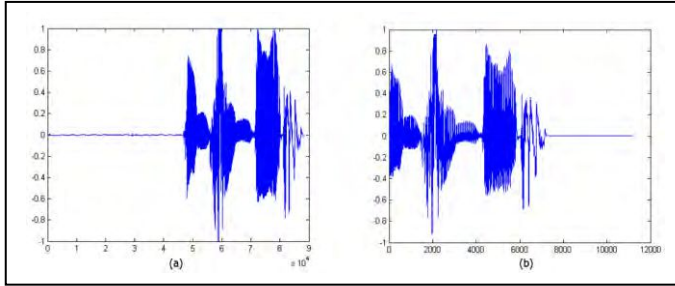


Figura 1. Comparación entre archivos de audio original y modificado (3.a. original; 3.b. modificado)

En la figura 1 se observan tres aspectos interesantes: se desplaza la señal para eliminar un silencio largo que se observa al principio, se normaliza la amplitud entre (-1,1), y la señal resultante se completa con ceros con el fin de que todas las señales tengan igual número de componentes.

El pretratamiento en frecuencia consiste en obtener el espectro de la señal utilizando el método de Welch [10]. Este método permite encontrar la representación en el dominio del tiempo de una señal temporal, utilizando para ello la transformada rápida de Fourier; en los estudios anteriores sobre reconocimiento de comandos de voz ([7,12]) se recomienda utilizarlo por su buen desempeño. En la aplicación de éste método se realiza el solapamiento de segmentos en búsqueda de disminuir la varianza. Adicionalmente se utilizan ventanas a las que se les aplica la transformada discreta de Fourier para formar los descriptores del sistema clasificador [7]. Utilizando éste método pueden manipularse los datos en términos de resolución y varianza, permitiendo eliminar datos innecesarios.

Para obtener un periodograma modificado por cada secuencia que se tiene en una ventana de tiempo predefinida se usa la ecuación 1. Donde $w(n)$ es la ventana, $x_i(n)$ es la secuencia de datos (señal de voz) que se tiene en cuenta en la ventana, L es la longitud de la secuencia y U se describe en la ecuación 2. El proceso de *enventanado* se refiere a aplicar a la señal de voz (multiplicar) una función limitada en el tiempo conocida como ventana, de esta manera se puede realizar un análisis localizado.

$$\hat{S}_M^{(i)}(e^{jw}) = \frac{1}{LU} \left| \sum_{n=0}^{L-1} w(n)x_i(n)e^{-jnw} \right|^2 \quad (1)$$

$$U = \frac{1}{L} \sum_{n=0}^{L-1} |w(n)|^2 \quad (2)$$

La transformada Welch, se obtiene mediante el promedio de los periodogramas modificados, como se observa en la ecuación 3. K es el número de secuencias.

$$\hat{S}_w(e^{jw}) = \frac{1}{K} \sum_{i=0}^{K-1} \hat{S}_M^{(i)}(e^{jw}) \quad (3)$$

En total, para cada palabra pronunciada, se obtienen 50 valores, los más significativos, para cada comando, que corresponden a las entradas del sistema clasificador. Cada palabra tiene un patrón de señal que permite diferenciarlo de los demás. En la figura 2 se presenta el patrón en frecuencia de cada uno de los comandos a reconocer, indicando el lugar donde se estima que se encuentran representadas cada una de las letras.

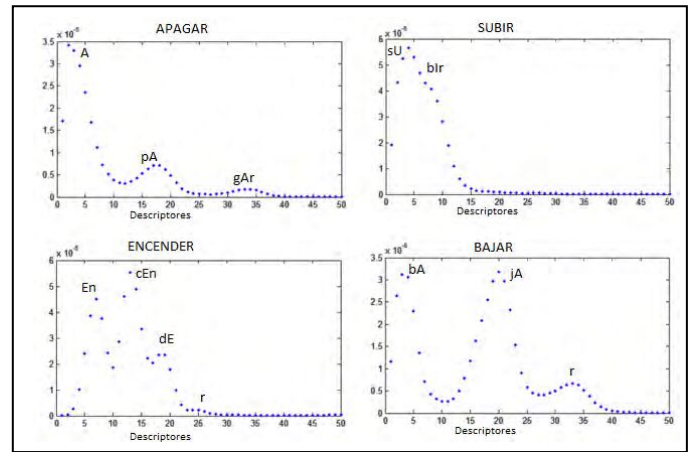


Figura 2. Patrones en frecuencia.

III. SISTEMA DE RECONOCIMIENTO DE VOZ

En esta sección se presentan los algoritmos probados con el fin de obtener un sistema de reconocimiento de voz robusto. En cada subsección se presentan los resultados obtenidos con cada algoritmo de reconocimiento de patrones que se utilizó.

A. Redes Neuronales

Una red neuronal artificial es un modelo que busca aproximarse al comportamiento de las neuronas biológicas, pueden aprender a diferenciar patrones con el fin de solucionar problemas complejos [6]. Cada neurona de la red se modela como un nodo que tiene diferentes entradas y una salida. Cada entrada de la neurona es ponderada por un peso. La salida de la neurona se calcula al evaluar, la sumatoria de las entradas (vector x_i) ponderadas por sus respectivos pesos (vector w_i), en una función de activación, proceso descrito en la ecuación 4.

$$y = F\left(\sum_i w_i \cdot x_i\right) \quad (4)$$

Los pesos son calculados por el algoritmo de entrenamiento de la red neuronal. Para esto, es necesario contar con datos históricos que ayudan a que el sistema identifique los patrones que luego, con datos no presentados anteriormente, debe reconocer. Al ajustar los pesos se busca que la salida de la red sea la esperada en todas las muestras, o en la mayoría de ellas, de tal forma que después de realizado el entrenamiento, la red deberá estar en capacidad de tener en la salida el valor esperado para una muestra totalmente nueva.

En el diseño de una red neuronal se distinguen dos etapas. La primera es la etapa de entrenamiento que permite escoger la estructura de la red, es decir, ajustar el número de capas y de neuronas por capa, además se encuentran los pesos óptimos. La segunda es la etapa de prueba que verifica la generalización del sistema.

Como se requiere reconocer comandos pronunciados por cualquier persona, independiente de su sexo o edad, es necesario incluir en los datos históricos audio de niños, adolescentes, jóvenes, adultos y personas de la tercera edad, de ambos sexos.

Para cumplir con lo anterior, se propone en la tabla I la distribución de los grupos de edades seleccionados para la recolección de muestras, buscando una mejor distribución para el aprendizaje. Los grupos mostrados en la tabla I, se establecieron tanto para hombres como para mujeres. Cada persona repitió 5 veces cada uno de los comandos a reconocer, de modo tal que por una sola persona se tienen 20 grabaciones.

TABLA I. CLASIFICACIÓN DE PERSONAS PARA RECOLECCIÓN DE MUESTRAS

Grupo	Rango de edad
1	Menores de 12 años
2	Entre 12 y 24 años
3	Entre 24 y 40 años
4	Mayores de 40 años

La estructura de la red neuronal usada para solucionar este problema consta de 50 descriptores (entradas, correspondientes a los valores tomados de la transformada de Welch) y 4 salidas.

Las entradas corresponden al resultado del pretratamiento, sección II, donde se obtienen las características más importantes de las muestras de audio. Las salidas se codifican para representar cada palabra, tal como se presenta en la tabla II.

TABLA II. VALORES DE LAS SALIDAS DE LA RED NEURONAL

Comando	Salida 1	Salida 2	Salida 3	Salida 4
Encender	1	0	0	0
Apagar	0	0	0	1
Subir	0	1	0	0
Bajar	0	0	1	0

- *Péceptor con entrenamiento backpropagation*

Entre los algoritmos de entrenamiento usados en redes neuronales [11], uno de los más utilizados para aplicaciones de reconocimiento de voz es el *backpropagation* ([12], [13]). El

entrenamiento mediante algoritmo *backpropagation* consiste en escoger aleatoriamente un dato para entrenar la red, al igual que los pesos de cada entrada para cada una de las neuronas, así se obtiene la salida de cada neurona, esta salida se propaga a través de las siguientes capas hasta generar la salida de la red neuronal; este resultado se compara con la salida deseada generando un error. El proceso siguiente es propagar hacia atrás dicho error para actualizar los pesos de las neuronas; este procedimiento se repite hasta que se cumpla una condición de parada.

Para encontrar la estructura de la red neuronal que presente el mejor resultado, se realizan diversos procesos de entrenamiento: variación de la tasa de entrenamiento (μ), de la cantidad de neuronas de cada capa, cantidad de capas ocultas, criterios de parada, etc.

Con la ayuda de las herramientas de MatLab se realizaron diferentes pruebas y configuraciones buscando el menor error en entrenamiento. Con entrenamiento de tipo *backpropagation*, se obtuvieron aciertos superiores al 90%, a partir de 5 neuronas en la capa oculta. Se intentó mejorar el resultado aumentando neuronas y capas pero no se presentaron resultados significativos. En la tabla III se muestra el resultado de la variación del número de neuronas en la capa oculta realizando entrenamiento de tipo *backpropagation*.

TABLA III. VARIACIÓN DE NEURONAS EN LA CAPA OCULTA

Nº neuronas en la capa oculta	Error entrenamiento	Error prueba
5	2.9%	18.6667%
6	4%	18.3333%
7	24%	33.6667%
8	2.9%	18%
9	2.1%	19.3333%
10	1.7%	15.6667%
15	3.1%	19.3333%
20	2.4%	18.3333%
25	26.4%	38.6667%

- *Redes neuronal de Base Radial*

Las funciones de base radial se caracterizan porque su salida depende de la distancia a un punto denominado centro, estas funciones son simétricas respecto a $x=0$.

Las neuronas de la capa oculta reciben las señales de la capa de entrada y realizan una transformación local sobre dichas señales. Esta transformación es realizada por la función de base radial. Cada elemento de la capa oculta representa una clase o categoría determinada por (c_i, d_i) , donde c_i es el centro del grupo interpretado como los pesos asociados a cada neurona i , y d_i representa la desviación o ancho de la función de base radial.

La salida de cada neurona de la capa oculta, consiste en aplicar una función de base radial al resultado de calcular la distancia entre la entrada y el centro de la función y ponderar este valor con $1/d_i$ (ecuación 5).

$$z_i(n) = \left(\frac{\sum_{j=1}^p \sqrt{(x_j(n) - c_{ij})^2}}{d_i} \right) \quad (5)$$

Donde z_i es la salida de la neurona i , y x_j representa a las p entradas de la neurona i . La función de base radial más utilizada es la gaussiana descrita en la ecuación 6.

$$\varphi(r) = e^{-\frac{r^2}{2}} \quad (6)$$

Con una red neuronal de base radial se obtuvo mejor resultado al usar 50 neuronas en su capa de entrada, 10 en la capa oculta y 4 en la de salida. Los aciertos en la etapa de entrenamiento fueron cercanos al 86% y en prueba 80%.

B. Máquinas de soporte vectorial

Las máquinas de soporte vectorial SVM (por sus siglas en inglés, *Support Vector Machine*) es una técnica de clasificación relativamente nueva que ha mostrado ser muy efectiva en problemas de clasificación binaria (dos clases) [10].

Esta técnica consiste en realizar una transformación del espacio de las entradas del sistema, a un espacio de dimensión superior donde se tiene la representación lineal de un sistema no lineal. Esto facilita la clasificación. Con el objetivo de obtener la representación de los datos originales en un espacio lineal, se utiliza una función conocida como *kernel*.

La capacidad de generalización de las máquinas de soporte vectorial se basa en el principio de minimización de riesgo estructural (SRM) que depende de un parámetro llamado dimensión VC, el cual se puede optimizar al mismo tiempo que se obtiene el menor error de entrenamiento [5].

En el entrenamiento lo que se busca es maximizar la distancia entre los datos y el hiperplano que los separa, esto se logra maximizando la función de costo $Q(\alpha)$ (ecuación 7).

$$Q(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j d_i d_j K(x_i, x_j) \quad (7)$$

Donde N es el número de muestras (número total de datos), α son los multiplicadores de LaGrange, d es la salida correspondiente al subíndice (i o j) y K es la función *kernel*, ésta última es seleccionada por criterio de quien soluciona el problema.

Para entrenar el sistema usando máquinas de soporte vectorial, se probaron tres tipos de funciones *kernel*, a saber, lineal, polinómica y gaussiana. Con el lineal y el polinómico se obtuvieron errores de entrenamiento superiores al 60% y por más que se ajustaron los parámetros de cada uno de ellos no fue posible disminuir dicho error.

Con el *kernel* gaussiano el error de entrenamiento disminuyó considerablemente, y al realizar un ajuste de cada uno de los parámetros de entrenamiento, propios de este tipo de *kernel*, se obtuvo un error de entrenamiento de 2.67%. Sin embargo el error de prueba se mantuvo por encima del 60%.

Lo anterior permite afirmar que el sistema se sobre-entrena y no logra la generalización, pues al ingresarle datos que no han participado en la etapa de entrenamiento, se presentan muchos errores. Para comprobar esto, se realizaron dos pruebas más.

La primera de las pruebas consistió en entrenar el sistema con todos los datos disponibles (1300), como resultado se obtuvo un error de 2.92%. Esto muestra que el sistema tiene la capacidad de adaptarse muy bien a los datos que son presentados en la etapa de entrenamiento.

La segunda prueba consistió en entrenar con 130 datos (los datos usados normalmente en el conjunto de prueba) y realizar la etapa de prueba con 1170 datos (los datos normalmente usados para entrenamiento). Como resultado se obtuvo un error de entrenamiento de 0% y un error de prueba superior al 60%.

Las pruebas realizadas demuestran la capacidad del sistema de adaptarse a los datos de entrenamiento, y la dificultad para realizar generalización.

IV. SISTEMA DE CONTROL DE INTENSIDAD DE LUZ

El sistema para control de intensidad de luz que se propone se representa con el esquema de la Figura 3. Se observa que el sistema se constituye de 6 etapas básicas, captura de muestras, pre tratamiento en tiempo y frecuencia, reconocimiento de voz, que de acuerdo a los resultados presentados en la sección anterior se realiza con una RNA tipo pércptron multicapa, después viene una etapa de comunicaciones que permite transmitir el comando identificado y una etapa de potencia que regula, de acuerdo a la señal de control, la energía suministrada a la lámpara.

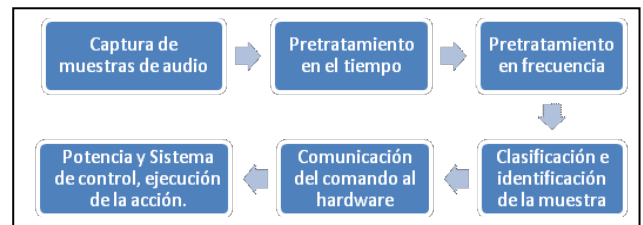


Figura 3. Diagrama de bloques del sistema

El hardware diseñado debe permitir la comunicación y regular la cantidad de energía que se le entrega a la lámpara. Lo anterior implica el diseño de: una fuente de alimentación, un sistema capaz de realizar comunicación con el computador y un circuito que permita regular la intensidad de luz de la lámpara. En la Figura 4 se presenta un esquema del hardware diseñado.

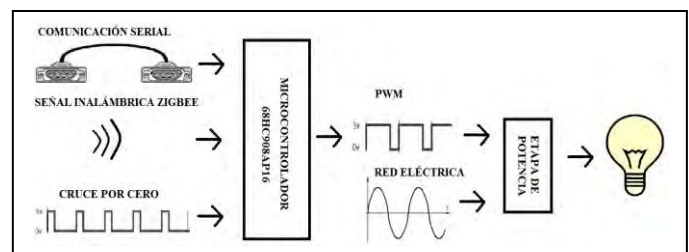


Figura 4. Esquema del hardware diseñado.

A. Comunicación

La comunicación consiste en enviar desde el computador una trama hacia el hardware externo, donde se informe cual fue el comando reconocido por la RNA. Para construir esta trama

se usaron 2 bytes, el primero de ellos es para indicar el inicio de la comunicación (I) y el segundo contiene la letra inicial de la palabra detectada (e=encender, a=apagar, s=subir, b=bajar).

El hardware diseñado permite fácilmente que la comunicación sea con cable o inalámbrica según la necesidad del usuario. En el primer caso se debe conectar un cable serial entre el computador y el hardware externo para establecer la comunicación. En el segundo caso se utiliza el protocolo ZigBee [14] basado en el estándar IEEE 802.15.4, el cual brinda seguridad en la comunicación y un bajo consumo de energía, ideal para el uso de la domótica.

Se decidió realizar la comunicación utilizando la tecnología Zigbee, basándose en la tabla IV, donde se comparan diferentes alternativas de comunicación.

Para establecer la comunicación inalámbrica mediante el protocolo ZigBee se requiere de dos dispositivos, uno de ellos se debe configurar como coordinador y conectarse al computador, este es el encargado de enviar la señal de información que contiene la acción a realizar sobre la lámpara; el segundo dispositivo se conecta al hardware externo y se configura como dispositivo final o como *router*.

TABLA IV. COMPARACIÓN ENTRE TECNOLOGÍAS DE COMUNICACIÓN INALÁMBRICA

Estándar	Ancho de banda	Consumo de potencia	Ventajas	Aplicaciones
Wi-Fi	Hasta 54 Mbps	400 mA transmitiendo 20 mA en reposo	Gran ancho de banda	Navegar por Internet, transferencia de ficheros
Bluetooth	1 Mbps	40 mA transmitiendo 0.2 mA en reposo	Interoperatividad (sustituto del cable)	Wireless USB, móviles, Informática casera
Zigbee	250 Kbps	30 mA transmitiendo 3mA en reposo	Batería larga duración. Bajo costo	Control remoto, sensores, juguetería

B. Detector de cruce por cero

En la figura 5 se observa en qué consiste la señal de cruce por cero. Mediante una comparación con un nivel DC, idealmente 0V, se determina el momento en que la señal de la red eléctrica cruza ese nivel. La señal de cruce por cero debe ir a la entrada IRQ (*Interrupt Request*) del microcontrolador para garantizar el sincronismo con la red eléctrica.

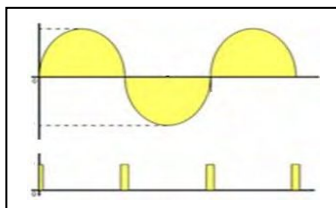


Figura 5. Detección del cruce por cero

C. Microcontrolador

El microcontrolador tiene dos entradas, una para la recepción del comando enviado por el PC, y otra para la interrupción de IRQ. La salida del microcontrolador es una señal PWM (*Pulse Width Modulation*) que permite el control de potencia de la lámpara.

La tarea del microcontrolador consiste en recibir, por puerto serial, la trama enviada desde el computador e interpretarla para establecer la forma de onda PWM que se entregará al circuito de potencia. Por el puerto IRQ ingresa una señal que determina en qué momento la señal de la red eléctrica cruza por cero, a partir de ésta se determina la configuración del PWM para garantizar el sincronismo con la red eléctrica. En este prototipo, se utiliza el microcontrolador 68HC908AP16 que es de bajo costo.

El algoritmo utilizado para el microcontrolador está orientado a la atención de eventos o interrupciones. El microcontrolador se mantiene en modo de bajo consumo mientras no reciba interrupciones, las cuales pueden darse, por el puerto IRQ o por el de comunicación serial. Una interrupción por comunicación serial recibe la trama enviada desde el PC, la interrupción por IRQ se utiliza para saber el momento adecuado, sincronismo con la red eléctrica, para actualizar el ángulo de conducción.

D. Etapa de potencia

La etapa de potencia incluye un sistema de opto acoplamiento para proteger el microcontrolador, y el circuito de potencia controlado por un TRIAC (por sus siglas en inglés *Triode for Alternating Current*). La señal de salida del microcontrolador corresponde a un determinado ángulo de conducción, de 0° a 180° , en función del comando reconocido y del estado actual de la lámpara, de tal modo que permite variar la cantidad de energía que se entrega a la lámpara como lo muestra la figura 6.

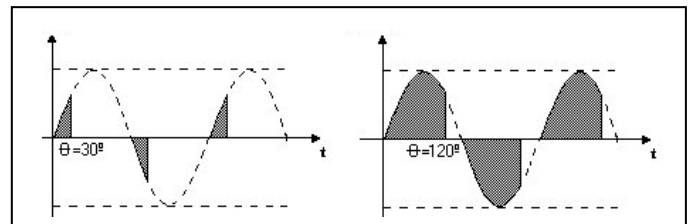


Figura 6. Ejemplos de la señal que alimenta a la lámpara

El control de potencia utilizado corresponde a la técnica de fase que manipula el ciclo útil de la señal de forma que entre mayor sea éste, mayor será la intensidad de luz [15].

V. RESULTADOS

El hardware diseñado para el funcionamiento del sistema, cumple con las especificaciones y el esquema ilustrados en la sección IV, y permite verificar el funcionamiento del sistema. A partir del entrenamiento se escoge una red neuronal de 10 neuronas en su única capa oculta y 4 en la de salida, con un acierto del 98.3% con los datos de entrenamiento (1170) y 84.4% en prueba (130).

Con el fin de validar los resultados de la red neuronal entrenada, se utilizó la técnica *k-fold cross-validation* [16], la cual consiste en dividir los datos en k particiones; para cada una de las cuales, se entrena la red con los datos de las $k-1$ particiones restantes y se realiza la prueba con la partición seleccionada. Particularmente se aplicó *10-fold cross-validation* es decir que el número de particiones en las que se dividen los datos es de 10. En la tabla V se presentan los resultados obtenidos al realizar esta validación.

TABLA V. RESULTADOS DE 10-FOLD CROSS-VALIDATION

Grupo	Error entrenamiento	Error prueba
1	3.5897%	14.6154%
2	2.1368%	19.2308%
3	1.2821%	16.9231%
4	3.5897%	16.9231%
5	1.7949%	16.1538%
6	3.5897%	18.4615%
7	2.7350%	14.6154%
8	3.0739%	11.5385%
9	2.4786%	18.4615%
10	5.0427%	14.6154%

Los resultados del *10-fold cross-validation* no difieren considerablemente respecto a la red seleccionada, mostrando que el sistema propuesto es capaz de generalizar.

VI. CONCLUSIONES Y PERSPECTIVAS

Se diseñó e implementó un sistema que permite reconocer, entre las palabras Encender, Bajar, Apagar y Subir, un comando de voz para controlar la intensidad de luz de una lámpara y responde a la voz de personas de diferente género y edad. Para la identificación del comando de voz se usa una RNA *perceptron* multicapa implementada en un computador, y la transmisión de la señal es inalámbrica usando la tecnología Zigbee. Este sistema se puede considerar un prototipo útil en aplicaciones de domótica.

Se comparó el desempeño de redes neuronales de base radial, *perceptron* multicapa con entrenamiento *backpropagation* y máquinas de soporte vectorial. El desempeño fue mejor para la red neuronal de tipo *perceptron* multicapa donde se demostró la generalización al utilizar la técnica *10-fold cross-validation*.

De los errores cometidos al identificar un comando de voz, se observó que la mayoría se presentan al pronunciar apagar y bajar esto se debe a las coincidencias fonéticas de dichas palabras. En la mayoría de los casos se tiende a pronunciar la primera *a* de apagar con un bajo volumen, lo que marca la similitud en las dos palabras mencionadas, con picos en el espectro para las dos *a* finales.

La orientación que se le da al sistema de reconocimiento de voz en este artículo es exclusivamente al área de la domótica, pero existen muchas otras áreas en las que se podría aplicar, debido a que la importancia real de este sistema está en

reconocer la voz de diferentes personas independiente del sexo y la edad. Incluso se probó el sistema usando como carga un ventilador obteniendo buenos resultados.

Los componentes de hardware funcionan correctamente y cumplen con las necesidades, sin embargo debe buscarse eliminar el computador como elemento de grabación e identificación del comando y reducir el tamaño del circuito de control. A futuro se buscará implementar las tareas del computador en un dispositivo de hardware compacto. Para el reconocimiento de comandos de voz, la red neuronal hace operaciones simples, de multiplicación y suma, y la evaluación de la función de activación. Siendo fácil de implementar en un microcontrolador. El reto principal se encuentra en la etapa de captura de la voz y la transformada que se le aplica, para esto puede utilizarse un procesador de señales o un microprocesador.

REFERENCIAS

- [1] CEDOM, "Instalaciones domóticas, cuaderno de buenas prácticas para promotores y constructores.", Segunda edición, Revista CEDOM, 2008.
- [2] José Luis O., "Algoritmos y Métodos para el Reconocimiento de Voz en Español Mediante Sílabas", Computación y sistemas, Vol. 9, 2006
- [3] Genoveva V. "SISTEMA DE RECONOCIMIENTO DE VOZ EN MATLAB," Universidad de San Carlos, 2008.
- [4] Simon H., Neural networks and learning machines, 3 ed. Prentice Hall, 2009.
- [5] José H., "Redes Neuronales Artificiales, Fundamentos, Modelos y Aplicaciones", Alfa Omega, 2000.
- [6] Andres N., "Redes Neuronales Artificiales", Universidad de la república, Uruguay, 2009
- [7] Ph.D. Cesar T. et al. "RECONOCIMIENTO AUTOMÁTICO DEL HABLA UTILIZANDO LA TRANSFORMADA DE FOURIER Y REDES NEURONALES.," Revista Colombiana de Tecnologías de Avanzada, vol. 2, 2005.
- [8] Germán M. et al. "Reconocimiento de comandos de voz con máquinas de soporte vectorial a través de bandas espectrales", Scientia et thechnique, Num.27, 2007.
- [9] Jorge M. et al. "Reconocimiento de comandos de voz usando la transformada wavelet y máquinas de vectores de soporte", Scientia et thechnique, Num.31, 2006.
- [10] "WELCH: PROMEDIADO DE PERIODOGRAMAS MODIFICADOS" [En línea].
<<http://physionet.cps.unizar.es/~eduardo/docencia/tds/librohtml/welch1.htm>> [consulta: 22 Enero 2011]
- [11] Luis B., "Entrenamiento de redes neuronales basado en algoritmos evolutivos", Universidad de Buenos Aires, 2005.
- [12] M.A. Acevedo et al. "Reconocimiento de Voz usando Redes Neuronales Artificiales Backpropagation y Coeficientes LPC." Sexto congreso internacional de computo en optimización y software, 2008.
- [13] Yves C. et al. Backpropagation: theory, architectures, and applications. Ed, 1, 1995
- [14] "Zigbee", [en línea] <<http://www.zigbee.org>> [consulta: 14 Octubre 2010]
- [15] Muhammad R., "Electrónica de potencia: circuitos, dispositivo y aplicaciones" Tercera edición, Pearson educación, México 2004.
- [16] Payam R. et al. "Cross - Validation", [en línea] <<http://www.public.asu.edu/~ltang9/papers/ency-cross-validation.pdf>> [consulta: 10 de octubre de 2010]